TELKOMNIKA, Vol.16, No.3, June 2018, pp. 1376~1385
ISSN: 1693-6930, accredited **A** by DIKTI, Decree No: 58/DIKTI/Kep/2013
DOI: 10.12928/TELKOMNIKA.v16i3.7288

Application Development Risk Assessment Model Based on Bayesian Network

Jaka Sembiring*[1], Johan P. Sipayung[2], Arry A. Arman[3]
[1,3]Institut Teknologi Bandung, School of Electrical Engineering and Informatics
[2]BPS, Statistics of Tapanuli Utara Regency
[1]Jl. Ganesha 10 Bandung 40132, Indonesia
*Corresponding author, e-mail: jaka@itb.ac.id

Abstract

This paper describes a new risk assessment model for application development and its implementation. The model is developed using a Bayesian network and Boehm's software risk principles. The Bayesian network is created after mapping top twenty risks in software projects with interrelationship digraph of risk area category. The probability of risk on the network is analyzed and validated using both numerical simulation and subjective probability from several experts in the field and a team of application developers. After obtaining the Bayesian network model, risk exposure is calculated using Boehm's risk principles. Finally, the implementation of the proposed model in a government institution is shown as a real case illustration.

Keywords*: bayesian network, risk assessment, sei taxonomy-based identification, risk area category, risk exposure*

1. Introduction

To develop a high quality application on time and within budget, one usually has to deal with various risks [1]. It is widely known that the success rate of successful IT project is very low. This fact is supported by a research from Standish Group on IT project 1994-2004 where it is shown that the completely failure rate was 18% in all projects, 53% of them were completed with unsatisfactory time, costs or effect, and only 29% of them had successfully accomplished the project target [2]. These facts show us that it is necessary to consider risk assessment as a way to systematically identify whether the occurrence of risk may affect the objectives of organization. Risk assessment method, as part of risk management, can be used as a tool to analyze both opportunities and consequences of risk prior to decide the next strategic action [3-4]. Risks in application development are related not only to the resources and functional problems encountered in the process of developing application but also to the impact of such problems.

Risk assessment in application development is a process to identify risk factors such as lack of clarity of project requirements, delivery not according to schedule and time, and failure in achieving the main objective of application development project [5]. Risk assessment procedure generally consists of risk identification, risk analysis and risk evaluation; and it provides an understanding of risks, their causes, consequences, and their probabilities [3-4]. According to Tao, risk management in software projects should focus on prevention and reduction of risks, assessing the likelihood of problems, determining risk potential that could become a major concern [6]. In other development, Sonchan et.al. have introduced top 20 software risks based on the frequency of their citations on highly referred and recent literature in the area of risk management of application development projects. They have succeeded in extracting and classifying top risks from thirty most frequently cited and recently published literatures on software project risks. They used Delphi method to propose potential impacts and probabilities of all classified risk [7]. They categorize these risks based on risk taxonomy described in [8]. However, their study did not investigate the interrelationship and probability of risks.

Other study from Gallagher has identified risk areas that are interconnected and he used interrelationship digraph to identify the cause and effect between risks [9]. He describes risk areas involving: (i) schedule pressure and veracity, (ii) suppression of information, (iii)

Received September 14, 2017; Revised February 14, 2018; Accepted March 24, 2018

requirement management, (iv) facility funding, (v) people, resources and leadership, (vi) operability, (vii) reliability and dependability, and (viii) testing.

The study in [7] does not cover the possibility of relation between risks, where the study in [8] indicates that it is possible to find the relationship between risks. Based on these research result, we enhance the work of [7] to show the possibility of relation between the top 20 risks. Earlier effort on this topic can be seen in [10] where the theoretical background, context diagram, mapping table, and preliminary model were elaborated. Our paper investigates further on how to determine the probability and interrelationship of such risk in application development using Bayesian network concept. Moreover, this paper also presents the impact of such risk using Boehm's application risk principles and risk exposure. Finally the implementation of the proposed model as a real case illustration in a government agency will be elaborated.

2. Research Method

This section will describe the step-by-step procedure to develop the proposed model. Initially, we adopt the research result on top twenty risks in software project as our basic risk classification [7]. In order to find the relationship among risks, those mentioned risks are mapped to the SEI Taxonomy-Base Risk Identification to find the characteristic of related risks [8]. Then based on the characteristic of the risks, the Category of Risk Area identified by Gallagher is used to find the interrelationship among risk [9]. The obtained grouping is shown in Table 1 to explain the relationship between risk areas. This result can also be seen in our previous research [10].

Table 1. Proposed mapping table of risk assessment model

Taxonomy-Based Risk Identification (Class)	Taxonomy-Based Risk Identification (Category)	Top Twenty Risks in Software Projects	Risk Area Identified
Product Engineering	Requirements	User resistance	Suppression of Information
Product Engineering	Requirements	Unable to meet user requirements	Reliability and Dependability
Product Engineering	Code and unit test	Low software performance	Testing
Development Environment	Development Process	Inappropriate development process	Operability
		Requirement creep	
Development Environment	Development System	Problems with new technology	
		Lack of technical skills	
		Technical complexity	
Development Environment	Management Methods	Unclear customer requirements	Requirements Management
		Optimistic resource planning	
		Inefficient team capability	
		Lack of executive involvement	
Development Environment	Management Process	Unrealistic budgeting	Facility Funding
		Unrealistic schedule	Schedule Pressure and Veracity
Development Environment	Work Environment	Communication gaps	People, Resources and Leadership
		Conflicts among team members	
		Staff turnover	
Program Constraints	Resources	Resource insufficiency	
		Inadequate infrastructure	
		Lack of law enforcement	

With this grouping, one can create a dependency model based on interrelation of risk category area shown in Figure 1. For clarity and consistency in further discussion, we assign each relevan risk with specific codes as seen in Table 2. Using the works described in [9], a direct relation between risk items can be constructed as an initial model shown in Figure 2. As we can see, the network contains many node dependencies. The validity of this initial dependency needs to be verified for implementation in a real environment. To evaluate node dependencies, we turn to expert judgment through discussion and survey. The experts involved in this step are application developers that we selected from application developer team in

Statistics Indonesia. Based on their experience, we are able to eliminate the relation among risks that are considered irrelevant. The result of model refinement can be seen in Figure 3 and Table 3. In the next section we will show that this model is valid to be implemented in a real application based on judgment from international experts in the field and application developers.

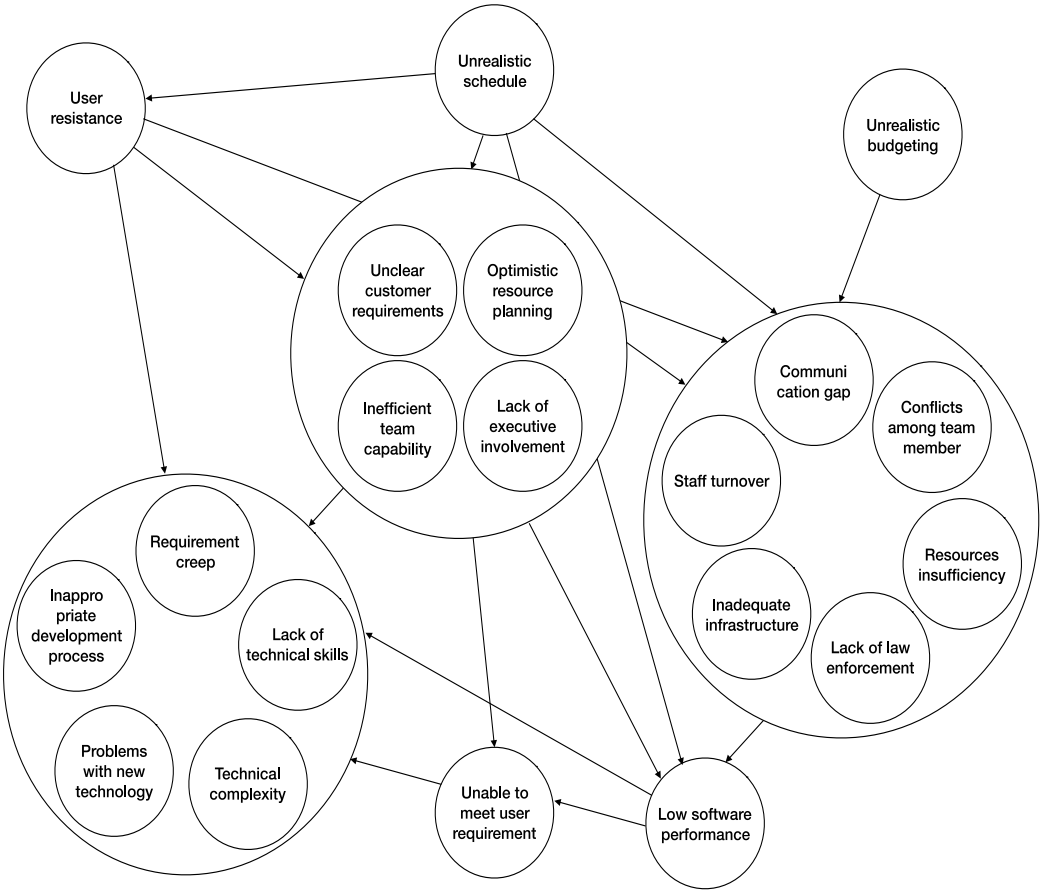

Figure 1. Proposed relationship model based on Category of Risk Area

Table 2. Codes for top twenty risk used in this paper

Code	Risk Item	Code	Risk Item
(1)	(2)	(3)	(4)
R1	Unclear customer requirements	R11	Unrealistic schedule
R2	Requirement creep	R12	Optimistic resource planning
R3	Unable to meet user requirements	R13	Lack of executive involvement
R4	Lack of technical skills	R14	Communication gaps
R5	Technical complexity	R15	Conflicts among team members
R6	Low software performance	R16	Staff turnover
R7	Inefficient team capability	R17	Unrealistic budgeting
R8	Inappropriate development process	R18	Resource insufficiency
R9	Problems with new technology	R19	User resistance
R10	Inadequate infrastructure	R20	Lack of law enforcement

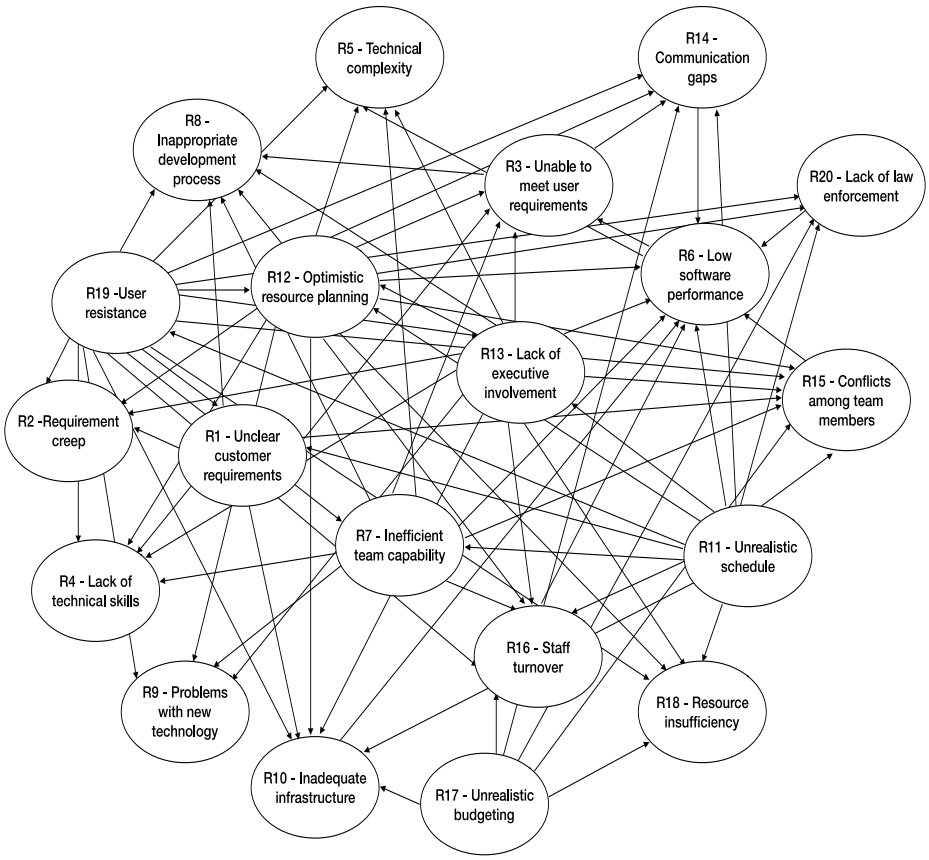

Figure 2. Initial Bayesian network model

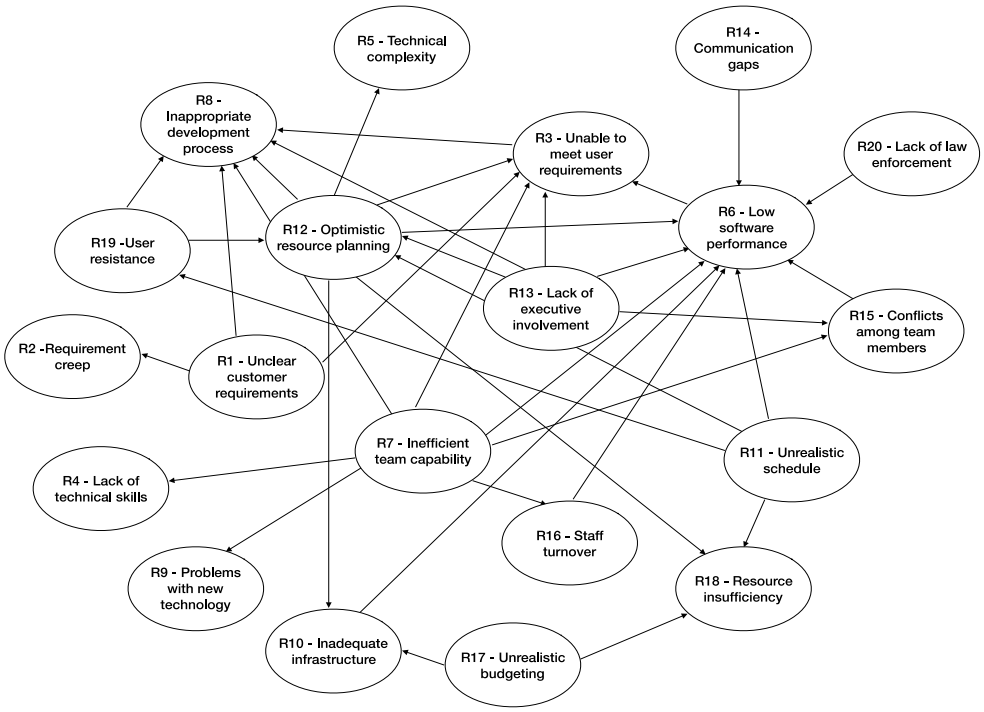

Figure 3. Refined and verified Bayesian network model

Table 3. Relationship among top twenty risk

Code	Risk Item (Cause)	Code	Risk Item (Effect)
(1)	(2)	(3)	(4)
		R2	Requirement creep
R1	Unclear customer requirements	R3	Unable to meet user requirements
		R8	Inappropriate development process
R3	Unable to meet user requirements	R8	Inappropriate development process
R6	Low software performance	R3	Unable to meet user requirements
		R3	Unable to meet user requirements
		R4	Lack of technical skills
		R6	Low software performance
R7	Inefficient team capability	R8	Inappropriate development process
		R9	Problems with new technology
		R15	Conflicts among team members
		R16	Staff turnover
R10	Inadequate infrastructure	R6	Low software performance
		R6	Low software performance
R11	Unrealistic schedule	R12	Optimistic resource planning
		R18	Resource insufficiency
		R19	User resistance
R12	Optimistic resource planning	R3	Unable to meet user requirements
		R5	Technical complexity
		R6	Low software performance
		R8	Inappropriate development process
		R10	Inadequate infrastructure
		R18	Resource insufficiency
R13	Lack of executive involvement	R3	Unable to meet user requirements
		R6	Low software performance
		R8	Inappropriate development process
		R12	Optimistic resource planning
		R15	Conflicts among team members
R14	Communication gaps	R6	Low software performance
R15	Conflicts among team members	R6	Low software performance
R16	Staff turnover	R6	Low software performance
R17	Unrealistic budgeting	R10	Inadequate infrastructure
		R18	Resource insufficiency
R19	User resistance	R8	Inappropriate development process
		R12	Optimistic resource planning
R20	Lack of law enforcement	R6	Low software performance

In this paper, following model derivation, we work further for implementation. We expand the model to cover risk exposure and its impact on organization. We begin with exposure concept from Tan to find the risk factor [11]. Based on this risk factor we calculate the risk exposure using the result from Boehm [12]. The risk exposure is defined as the relation between probability of unsatisfactory outcome and loss as a consequence of those unsatisfactory outcomes. In this paper we use risk factor as an input of loss as a consequence of unsatisfactory outcome. We will use the following equation.

$$RE = \text{Prob}(UO) \times \text{Loss}(UO) \quad (1)$$

Where:

RE = Risk Exposure
Prob(*UO*) = Probability of unsatisfactory outcome
Loss(*UO*) = Loss as an impact of unsatisfactory outcome

Finally we use the Risk-Level matrix from Stoneburner to classify the risk. The risk calssification consists of three levels, i.e.: High, Medium, and Low [13]. With these levels, it will be easier for the management to comprehend the risk as a whole and to take action as a response to the risk.

3. Evaluation, Implementation and Discussion

3.1. Evaluation

To evaluate the model, first we perform a numerical simulation on several nodes. Since the model uses the basic concept of probability, then the following conditions described in equation (2) must be fulfilled.

$$0 \leq CPT \leq 1 \quad (2a)$$

$$P(S) = 1 \quad (2b)$$

$$0 \leq P(R) \leq 1 \quad (2c)$$

Where:

CPT probability in Conditional Probability Table
P(S) total probability at a specified node
P(R) probability of the risk at nodes using the proposed model

For illustration purpose we calculate the probability of node (R1|R2), which means the probability of "Unclear Customer Requirements" with evidence of risk "Requirement Creep". Assumed at random that the probability of node R1 is shown in Table 4, and assumed that the prior probability of (R1|R2) is shown in Table 5.

Table 4. Probability of R1

R1	T	F
Probability	0,7	0,3

Table 5. Probability of (R2|R1)

R1	P(R2|R1 = T)	P(R2|R1 = F)
T	0,9	0,1
F	0,7	0,3

Using simple probability theory we can calculate:

1. P(R2|R1=T)=0.9
   The risk's probability of "Requirement Creep" if risk of "Unclear Customer Requirements" occurs is 0.9.
2. P(R1)=0.7
   The risk's probability of "Unclear Customer Requirements" is 0.7.

Based on CPT in equation (2).

$$\begin{aligned} P(R2) &= (P(R2|R1=T) * (P(R1=F)) + (P(R2|R1=F) * (P(R1=T)) \\ &= (0.9*0.3) + (0.7*0.3) \\ &= 0.84 \end{aligned}$$

So that,

$$\begin{aligned} P(R1|R2) &= (P(R2|R1)*P(R1))/(P(R2)) \\ &= ((0.9*0.7))/0.84 \\ &= 0.75 \end{aligned}$$

We conclude that the risk's probability of "Unclear Customer Requirements" with evidence of risk of "Requirement Creep" is 0.75. We can calculate the probability of each node in the network with the same procedure and will obtain the consistency that each node fulfills the condition in equation (2).

In addition to evaluation by simulation, we also evaluate the model through judgment from international experts and application developers. We involve 20 application developers from the government agencies we mentioned before and 10 international researchers/experts in the field of application development. The result of relationship evaluation from 20 application developers and 10 researchers point of view can be seen in Table 6. Based on numerical simulation and subjective judgment in Table 6, we conclude that our proposed model has consistency and can be used as an assessment tools.

Table 6. Evaluation of relationship among risks based on point of view from 20 application developers and 10 researchers

Risk Item (Cause)	Risk Item (Effect)	Eval. of Relationship (Yes/No)	Appl. Dev. 'Yes' (%)	Expert 'Yes' (%)
	Requirement creep	Yes	80	60
Unclear customer requirements	Unable to meet user requirements	Yes	90	85
	Inappropriate development process	Yes	50	85
Unable to meet user requirements	Inappropriate development process	Yes	50	90
Low software performance	Unable to meet user requirements	Yes	50	85
	Unable to meet user requirements	Yes	60	65
	Lack of technical skills	Yes	50	75
	Low software performance	Yes	80	25
Inefficient team capability	Inappropriate development process	Yes	70	70
	Problems with new technology	Yes	30	20
	Conflicts among team members	Yes	90	70
	Staff turnover	Yes	50	50
Inadequate infrastructure	Low software performance	Yes	70	95
	Low software performance	Yes	60	50
Unrealistic schedule	Optimistic resource planning	Yes	70	75
	Resource insufficiency	Yes	80	45
	User resistance	Yes	10	25
	Unable to meet user requirements	Yes	40	55
	Technical complexity	Yes	50	55
Optimistic resource planning	Low software performance	Yes	60	50
	Inappropriate development process	Yes	60	60
	Inadequate infrastructure	Yes	40	35
	Resource insufficiency	Yes	80	75
Lack of executive involvement	Unable to meet user requirements	Yes	60	55
	Low software performance	Yes	10	20
	Inappropriate development process	Yes	40	60
	Optimistic resource planning	Yes	70	70
	Conflicts among team members	Yes	60	45
Communi-cation gaps	Low software performance	Yes	40	60
Conflicts among team members	Low software performance	Yes	40	75
Staff turnover	Low software performance	Yes	60	65
Unrealistic budgeting	Inadequate infrastructure	Yes	60	90
	Resource insufficiency	Yes	90	75
User resistance	Inappropriate development process	Yes	80	95
	Optimistic resource planning	Yes	40	35
Lack of law enforcement	Low software performance	Yes	40	45

3.1. Implementation and Discussion

For case illustration, we implement our proposed model for risk assessment in a government agency Statistics Indonesia. In our proposed model there are total 469 parameters. For implementation simplicity we use around 16% of the total variables to be implemented with real values. We are using subjective probability method to assess the selected value involving 20 application developers to obtain the probability value. Those values can be seen in Table 7 column (3). With the same respondents, using Tan method [11] to obtain exposure factor, we get the results given in Table 7 column (4).

Tabel 7. Risk and exposure factor values from 20 application developers view

Code	Risk	Average Probability	Average Exposure Factor
(1)	(2)	(3)	(4)
R1	Unclear customer requirements	0.66	47
R2	Requirement creep	0.69	37
R3	Unable to meet user requirements	0.48	54
R4	Lack of technical skills	0.36	57
R5	Technical complexity	0.46	60
R6	Low softw are performance	0.42	52
R7	Inefficient team capability	0.36	46
R8	Inappropriate development process	0.50	55
R9	Problems w ith new technology	0.32	49
R10	Inadequate infrastructure	0.35	61
R11	Unrealistic schedule	0.58	37
R12	Optimistic resource planning	0.50	51
R13	Lack of executive involvement	0.34	47
R14	Communication gaps	0.37	48
R15	Conflicts among team members	0.27	55
R16	Staff turnover	0.17	50
R17	Unrealistic budgeting	0.39	51
R18	Resource insufficiency	0.38	50
R19	User resistance	0.38	48
R20	Lack of law enforcement	0.30	55

From Tabel 7 column (3), we understood that in application development at Statistics Indonesia, the risk of R2 “Requirement Creep” has the highest value. In many cases users quite often asking to add new function or change user requirement during application development process where they are not able or do not explain the requirement in detail. This result in higher value of risk in user requirement. When we analyzed further the exposure factor from Table 7 column (4), the highest exposure are R10 “Inadequate infrastructure” (61) and R5 “Technical complexity” (60). It means that although the risk for both factors are not as high as R2 “Requirement Creep”, but the perceive impact of those risks to the organization are severe.

For the purpose of simplicity without sacrificing the integrity of the model, we select several conditional probabilities that we use for implementation of case illustration as shown in Table 8 column (1), where P(R1|R2) means “The probability of R1 given evidence R2”, P(R7|R3,R4,R6,R8,R9,R15,R16) means “The probability of R7 given evidence R3, R4, R6, R8, R9, R15 and R16” and so on.

Using our proposed model, all of the result of implementation can be seen in Table 8. In Table 8 column (2) we show the subjective probability from 20 application developers, and column (3) is the exposure factor derived from Table 7. The risk exposure in column (4) is calculated using Boehm’s formula in equation (1). Then using risk level from Stoneburner [13], we obtain the risk level of the selected risks in Statistics Indonesia as our object of implementation, which are shown in Table 8 column (5).

From Table 8 column (2) we obtain that the conditional risk P(R1|R2), which is the probability of “Unclear user requirement” provided that “Requirement creep” is very high reaching 0.73. This is consistent with our previous examination that both risks have the highest probability, and the combination of both resulted in higher number of probability of risk. When we assess the risk exposure further, we obtain that lower probability risk does not mean lower impact. We can see that the impact of P(R1|R2), P(R1|R2,R3), P(R12|R3,R5,R6,R8,R10), and P(R12|R3,R5,R6,R8,R10,R18) are higher even though they have different probability risk. It is clear that the perceived exposure factor or impact has played significant role in this result. When

we calculate the risk level using Stoneburner method, it happened that in our case illustration all of the risk levels are “Medium”, in other circumstances at other organization the results could have been different. Using our proposed method, we have shown that it will be easier for the management and application developers in organization to assess the risks and their impact on organization so that it will help them to create a policy or do some actions to anticipate and manage the risks.

Table 8. Implementation result of our proposed model

Probability	Avrg. Value of Probability	Avrg. Value of Exposure Factor	Avrg.Value of Risk Exposure	Risk Level
(1)	(2)	(3)	(4)	(5)
P(R1|R2)	0.73	47	35	M
P(R1|R2,R3)	0.72	47	34	M
P(R1|R2,R3,R8)	0.68	47	32	M
P(R3|R8)	0.49	54	26	M
P(R6|R3)	0.50	52	27	M
P(R7|R3)	0.39	46	17	M
P(R7|R3,R4)	0.48	46	21	M
P(R7|R3,R4,R6)	0.50	46	22	M
P(R7|R3,R4,R6,R8)	0.45	46	20	M
P(R7|R3,R4,R6,R8,R9)	0.51	46	23	M
P(R7|R3,R4,R6,R8,R9,R15)	0.58	46	27	M
P(R7|R3,R4,R6,R8,R9,R15,R16)	0.64	46	30	M
P(R10|R6)	0.51	61	30	M
P(R11|R6)	0.60	37	21	M
P(R11|R6,R12)	0.62	37	22	M
P(R11|R6,R12,R18)	0.63	37	23	M
P(R11|R6,R12,R18,R19)	0.57	37	20	M
P(R12|R3)	0.58	51	29	M
P(R12|R3,R5)	0.65	51	33	M
P(R12|R3,R5,R6)	0.66	51	33	M
P(R12|R3,R5,R6,R8)	0.65	51	33	M
P(R12|R3,R5,R6,R8,R10)	0.67	51	34	M
P(R12|R3,R5,R6,R8,R10,R18)	0.68	51	34	M
P(R13|R3)	0.37	47	18	M
P(R13|R3,R6)	0.39	47	19	M
P(R13|R3,R6,R8)	0.40	47	19	M
P(R13|R3,R6,R8,R12)	0.45	47	22	M
P(R13|R3,R6,R8,R12,R15)	0.46	47	23	M
P(R15|R6)	0.38	55	21	M
P(R16|R6)	0.25	50	12	M
P(R17|R10)	0.39	51	20	M
P(R17|R10,R18)	0.43	51	21	M
P(R19|R8)	0.35	48	17	M
P(R19|R8,R12)	0.27	48	12	M

Note: M (Medium Risk Level)

4. Conclusion

In this paper, we propose a risk assessment model of application development using Bayesian network and Boehm's software risk principles. After classifying, mapping and converting all risks to network nodes, we obtained the interrelationships among risks of top 20 risks in application development projects. Then we calculate the risk exposure on the derived network. By implementing the method in a real environment in a government agency, we conclude that management level and application developers can use our proposed risk assessment model to evaluate relationship, calculate probability and impact of risks to manage risk properly and appropriately.

References

[1] Westfall L. Software Risk Management. The Westfall Team. 2001.
[2] Xu F, Qi G, Sun Y. The Risk Analysis of Software Projects Based on Bayesian Network. *Journal of Convergence Information Technology*. 2012; 7(5).
[3] IEC. International Standard IEC/ISO 31010. Risk Management-Risk Assessment Techniques. 2009.
[4] Pandey SK, Mustafa K. A Comparative Study of Risk Assessment Methodologies for Information Systems. *Bulletin of Electrical Engineering and Informatics (BEEI)*. 2012; 1(2): 111-122.
[5] Ai-Guo T., Ru-Long W. *Software Project Risk Assessment Model Based on Fuzzy Theory.* International Conference on Computer and Communication Technologies in Agricultural Engineering. 2010: 328-330.
[6] Tao Y. *A Study of Software Development Project Risk Management.* International Seminar on Future Information Technology and Management Engineering. 2008: 309-312.
[7] Sonchan P, Ramingwong S. *Top Twenty Risks in Software Projects: A Content Analysis and Delphi Study.* The 11th Conference on Electrical Engineering/Electronics, Computer, Telecommunicationa and Information Technology. 2014: 1-6.
[8] Carr MJ, Konda SL, Monarch I., Ulrich F.C., Walker C.F. Taxonomy-Based Risk Identification. CMU/Software Engineering Institute. 1993.
[9] Gallagher B. A Taxonomy of Operational Risk. CMU/Software Engineering Institute. 2005.
[10] Sipayung JJP, Sembiring J. *Risk Assessment Model of Application Development using Bayesian Network and Boehm's Software Risk Principles.* International Conference on Information Technology, Systems and Innovation. 2015: A-50
[11] Tan D. Quantitative Risk Analysis Step-By-Step. SANS Institute. 2003.
[12] Boehm BW. Software Risk Management: Principles and Practices. *IEEE Software.* 1991; 8(1): 32-41.
[13] Stoneburner G, Goguen A, Feringa A. Risk Management Guide for Information Technology Systems. National Institute of Standards and Technology. 2002.